

UNITED STATES PATENT APPLICATION FOR

A FAST PIPELINED ADDER/SUBTRACTOR USING
INCREMENT/DECREMENT FUNCTION WITH REDUCED REGISTER
UTILIZATION

Inventors:

HANEEF D. MOHAMMED

LARRY HEMMERT

Prepared by:

WAGNER, MURABITO & HAO LLP

Two North Market Street

Third Floor

San Jose, California 95113

(408) 938-9060

A FAST PIPELINED ADDER/SUBTRACTOR USING INCREMENT/DECREMENT FUNCTION WITH REDUCED REGISTER UTILIZATION

FIELD OF THE INVENTION

5

Embodiments of the present invention relate to the field of arithmetic circuitry employing carry/borrow logic. More specifically, embodiments of the present invention pertain to a fast pipelined adder/subtractor with reduced register utilization.

10

Prior Art

15

It is understood that one can increase the throughput of combinatorial designs by breaking up an arithmetic function into discrete functions and adding registers at each discrete function to "partition" the arithmetic function into smaller segments. These registers then become a pipeline to store the data until needed later in the pipeline. The tradeoff is that speed is gained at the cost of more complexity and stages of latency. Throughput is increased because several pipelines can operate simultaneously.

20

The classical approach to pipelining an adder is shown in Figure 1 (Conventional Art). The data from input bus 105 is broken up into several pipelines (4 in this example) and registers are placed to hold the data in the

pipeline between clock cycles. The first pipeline consists of an adder 111 with several single width registers (112-115) following. Because of the architecture of a programmable logic device, the adders and register following it can be combined in one level of logic (e.g., one logic block or "macrocell") exemplified by adder elements 110, 121, and 132. The second pipeline consists of a double width register 120, for two operands, followed by and adder element 121 and filled out with additional registers. This pattern continues until the last pipeline, which consists of several double width registers and a final adder. Data is clocked between the registers based on a system clock which thereby defines the pipestages of the pipeline.

The adders also generate a carry out bit if the result is greater than will fit in the result registers. The carry out bit is pipelined through the system by single bit carry bit registers 113, 124, and 135. The sum output is then collected from the 4 pipelines and the carry bit. Traditionally, the latency from input to output is divided by the number of pipe stages used. The addition of several pipelines that operate simultaneously is what makes the pipelined adder throughput greater than that of a purely combinatorial version.

The disadvantage of this conventional method is the necessity for double width registers throughout the pipelines in many pipestages. In PLD (Programmable Logic Device) type architectures, this becomes a big disadvantage because the number of registers is limited. In the traditional

pipelined architecture, this causes unwanted usage of registers and macrocells beyond what is necessary for the generation of the logic.

It would be advantageous then, to provide a system which combines the advantages of a pipelined adder/subtractor with a reduced register count. A further need exists for a system which meets the above need while providing faster processing speed and reduced energy utilization.

SUMMARY OF THE INVENTION

Accordingly, embodiments of the present invention provide a system which reduces the register count in a pipelined adder/subtractor circuit. The present invention also provides faster processing speed and requires less energy over conventional art devices of similar function. Embodiments include a pipelined arithmetic circuit that may be implemented on a PLD.

Embodiments of the present invention are directed to a fast pipelined adder/subtractor using increment/decrement functions with reduced register utilization. Embodiments of the present invention replace double width registers with incrementor elements, pipelined single width registers, and pipelined carry bits. This is made possible by positioning the adder elements at the first pipestage in each of the pipelines. Single width registers then are used to hold the results of the initial add/subtract operation. Single bit registers pipeline the carry bit from the adder and incrementor carry out to the next stage. The incrementor collects the sum from one of the adder elements, the pipelined carry bit from that adder element, and the carry bit from a previous stage adder and combines them to produce a new result and carry. This new result is passed along the pipeline to the output bus of the circuit. In this fashion, no double width busses or registers are required in between individual pipestages of the pipelines.

The present invention utilizes multiple pipelined adder elements that are coupled with an input bus. Each adder element performs an add/subtract operation upon a pair of operands and stores the result in a single width register. Any carry bits resulting from the increment/decrement or add/subtract operations are sent to a single bit register. In the next pipestage, an incrementor is coupled with the single width register of one of the adder elements, the carry bit register of that same adder element, and the carry bit register of the adder element of the preceding stage. The incrementor outputs a new result and carry bit. The carry bit is pipelined through all of the stages and the result is carried in single width registers through the pipeline to an output line.

One advantage of the present invention is that it maintains the high throughput of conventional art designs while reducing the need for double width registers. There are now no double registers. The register complexity is replaced by a simpler and smaller incrementor element and some single bit registers for the carry bits. This is a distinct advantage for the architectures found in PLD type devices which have fewer available resources than gate array resources. By reducing the area the design occupies, there are also advantages in circuit performance due to having a more compact arrangement with smaller connect distances resulting in smaller delays and reduced power utilization.

These and other objects and advantages of the present invention will become obvious to those of ordinary skill in the art after having read the

following detailed description of the preferred embodiments which are illustrated in the various drawing figures.

FIG. 1 is a perspective view of a preferred embodiment of a device in accordance with the present invention, showing a rectangular base with a central opening and a handle on the right side.

BRIEF DESCRIPTION OF THE DRAWINGS

The accompanying drawings, which are incorporated in and form a part of this specification, illustrate embodiments of the present invention and, together with the description, serve to explain the principles of the invention.

FIGURE 1 is a block diagram of a prior art pipelined adder/subtractor.

FIGURE 2 is a block diagram of a pipelined adder/subtractor as embodied by the present invention.

FIGURE 3 is a flowchart of a process 300 for using a pipelined adder/subtractor to perform add/subtract functions as embodied by the present invention.

FIGURE 4 is a block diagram of an incrementor device 400 utilized in embodiments of the present invention.

DETAILED DESCRIPTION

In the following detailed description of the present invention, a fast pipelined adder/subtractor using increment/decrement function with reduced register utilization, numerous specific details are set forth in order to provide a thorough understanding of the present invention. However, it will be obvious to one skilled in the art that the present invention may be practiced without these specific details. In other instances well known methods, procedures, components, and circuits have not been described in detail as not to unnecessarily obscure aspects of the present invention.

Figure 2 is a block diagram of the pipelined adder/subtractor circuit 200 of the present invention. For purposes of clarity, the following discussion will utilize the block diagram of Figure 2 in conjunction with flow chart 300 of Figure 3, to describe one embodiment of the present invention.

With reference to Figure 2 and to step 305 of Figure 3, N-bit operands are received into adder elements 211, 221, 231, and 241 on the first clock cycle. In the present embodiment, adder/subtractor 200 comprises input busses 205 (one for each operand) coupled to four pipelines 260-290 which perform the add/subtract function of the present invention. In other embodiments of the present invention, there can be any number of pipelines in adder/subtractor 200. Each of these pipelines has as its first stage, an adder element (e.g., adder

elements 210, 220, 230, and 240). The adder elements are comprised of an adder (e.g., adders 211, 221, 231, and 241) and a single width (e.g., N-bit) register (e.g., registers 212, 222, 232, and 242). Because of the architecture of programmable logic devices, the adders and registers of the adder elements can be combined in one macrocell of logic per bit. The adders are simply combinatorial adder/subtractors which are well known in the art. They provide the basic mathematical function of: $\text{Sum} = A + B$ plus a carry. The need for any double width registers is eliminated by placing the adder elements in the first stage of the pipeline.

With reference to Figure 2 and to step 310 of Figure 3, an add/subtract operation is performed in each of the adder elements in the first clock cycle. After receiving two N-bit operands from input bus 205, the adders perform an add/subtract function and output an N-bit result to their respective result registers (e.g., registers 212, 222, 232, and 242). The result registers are single width registers as only one N-bit result is output by the adders. Registers 214, 215, 228, 234, 244, and 246 are incorporated to hold the result from their respective adders or incrementors for latency while the other pipelines are performing their respective operations.

The adders also generate a carry out bit if the result is greater than will fit in the result registers. The carry out is a single bit which is sent to a single bit register (e.g., register 213) and held until it is sent to an incrementor. This carry

out bit is used by the incrementor (e.g., incrementors 225, 236, and 248) of the next stage as a carry in. The multiple carry bit registers coupled with adders 231 and 241 are to hold the carry bit for latency while other pipelines are performing their respective operations. The result and carry bit registers are D-type flip-flops or other storage elements capable of holding the value of their input data from one clock edge to the next.

With reference to Figure 2 and to step 315 of Figure 3, an N-bit result and two carry bits are combined in an incrementor on the second clock cycle for pipestage 270, the third for pipestage 280, and the fourth for pipestage 290. In the second stage of adder/subtractor 200, incrementor element 225 receives an N-bit result from register 222, a carry bit from register 213, and a carry bit from register 223. Because of the architecture of a programmable logic device, an incrementor and a register can be combined in one macrocell level (e.g., increment elements 224, and 238). This allows better utilization of the macrocells in device 200 than that of the conventional art devices. Furthermore, the construction of an incrementor requires relatively few macrocells as opposed to the double registers necessary in the prior art implementation of a pipeline adder.

The function of an incrementor utilized in embodiments of the present invention is shown in greater detail in Figure 4. For purposes of clarity, reference will be made to Figure 4 and to incrementor 225 of Figure 2.

Incrementor 400 is comprised of a simple incrementor 410 and an OR gate 430. Incrementor 400 receives a first result into incrementor 410 from an adder element in its respective pipeline (e.g., adder element 220 of Figure 2) as well as a first carry bit from a second adder element (e.g., adder element 210 of Figure 2) in a different pipeline. Adder 410 performs an increment operation and outputs a second result and, if necessary, a carry bit.

The carry bit is input into OR gate 430 along with the first carry bit from the adder element 220. OR gate 430 performs a Boolean OR operation and outputs a second carry bit to carry bit register 440 (e.g., carry bit register 227 of Figure 2). It is appreciated that in some embodiments of incrementor 400 the second result and second carry bit may be directly input into output bus 250 (e.g., incrementor 248 of Figure 2). In other embodiments of incrementor 400, the second result and the second carry bit are stored in pipelined registers (e.g., registers 226 and 227 of Figure 2).

The incrementors can perform their operation without adding any latency to the pipeline. Thus, in a two pipeline adder/subtractor, outputs from an incrementor are directly input into output bus 250 (e.g., incrementor 248) without adding any extra step of latency in the system. This pattern of pipelining the carry bit to the next stage continues until the last pipeline 290. The pipelines 260-290 then output their results on the last clock cycle and the carry bit from the incrementor of the last pipestage to create the final arithmetic result.

With reference to Figure 2 and to step 320, a second result and carry bit are output from an incrementor. The result and the carry bit can be held in registers or, if necessary, directly input into output bus 250.

5

With reference to Figure 2 and to step 325, a final arithmetic result is supplied. The final result is the combination of the result registers of each pipeline, the output of the incrementor in the last stage, and the carry bit from the incrementor in the last stage.

10

While the present invention has been described in reference to increment/decrement operations, it is not intended to be limited to these operations only. On the contrary, it is also applicable to any other circuitry that makes use of carry/borrow logic.

15

The preferred embodiment of the present invention, a fast pipelined adder/subtractor using increment/decrement function with reduced register utilization, is thus described. While the present invention has been described in particular embodiments, it should be appreciated that the present invention should not be construed as limited by such embodiments, but rather construed according to the following claims.

20